Ontologies for Web Services

Mark Dutra
Sandpiper Software, Inc.
Mdutra@Sandsoft.com
Definitions

• Knowledge representation means that knowledge is formalized in a symbolic form, that is, to find a symbolic expression that can be interpreted. – Klein and Methlie

• Knowledge engineering is the application of logic and ontology to the task of building computable models of some domain for some purpose. – John Sowa

• An ontology is a specification of a conceptualization. – Tom Gruber
What are Ontologies?

• An ontology specifies a rich description of the:
  – Terminology
  – Concepts
  – Relationships between the concepts
  – Rules

relevant to a particular domain or area of interest
Kinds of Ontologies

• An *Upper-Level* ontology defines the base concepts upon which other ontologies are created.
• A *domain* ontology, which we might also call a *classic* ontology, defines the terminology and concepts relevant to a particular topic or area of interest.
• A *process* ontology defines the inputs, outputs, constraints, relations, terms, and sequencing information relevant to a particular business process or set of processes.
• An *interface* ontology defines the structure and content restrictions relevant for a particular interface (*e.g.*, application programming interface (API), database, scripting language, etc.).
• A *role-based* ontology defines terminology and concepts relevant for a particular end-user (person or consumer application).
KR Domain Overview

• Early work on KR done by the Artificial Intelligence community
• KR domain has historically neglected common software engineering discipline
• Mapping between KR concepts and software engineering concepts not always straightforward
• Granularity of ontology models varies greatly between organizations
Why Not XML?

- Ontologies allow you to access and share more of your data.
- Ontologies improve accuracy, promote completeness.
- Ontologies are more flexible than simple tags.
- Ontologies provide semantics to the information “between the tags”
Use of Ontologies on the Web

- Improves the accuracy of web searches by searching for concepts instead of keywords.
- Allows systems that were independently developed to work together to exchange information.
- Facilitates the use of agents to collect, process, and exchange information.
- Helps tackle complicated questions whose answers do not reside on a single web page.
Ontology Development Issues

- Ontology methodology fragmentation – DAML should help
- Poor tool support
- Lack of existing ontology base
- Scale of ontologies – need a component-based approach
- Scope of ontologies – where do you stop?
Motivation to use UML for Ontology Development

- Importance of knowledge representation (ontologies) increasing
- No existing commercial tools for Ontology modeling
- Pool of experienced ontologists small
- Population of UML experienced engineers is growing
- Need to make Ontology modeling accessible to domain experts
## UML Profile for Ontologies

<table>
<thead>
<tr>
<th>Frame-Based KR (Ontology) Element</th>
<th>UML Metamodel Element(s)</th>
<th>UML Stereotype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ontology</td>
<td>Package</td>
<td>Ontology</td>
</tr>
<tr>
<td>Class</td>
<td>Package</td>
<td>ClassFrame</td>
</tr>
<tr>
<td>Class</td>
<td>Class</td>
<td>OntologyClass</td>
</tr>
<tr>
<td>Relation</td>
<td>Package</td>
<td>RelationFrame</td>
</tr>
<tr>
<td>Class</td>
<td>Class</td>
<td>Relation</td>
</tr>
<tr>
<td>Association</td>
<td>Association</td>
<td>Domain</td>
</tr>
<tr>
<td>Function</td>
<td>Package</td>
<td>FunctionFrame</td>
</tr>
<tr>
<td>Class</td>
<td>Class</td>
<td>Function</td>
</tr>
<tr>
<td>Association</td>
<td>Association</td>
<td>Domain</td>
</tr>
<tr>
<td>Association</td>
<td>Association</td>
<td>Range</td>
</tr>
<tr>
<td>Operation</td>
<td>Operation</td>
<td>Function</td>
</tr>
<tr>
<td>Individual</td>
<td>Class</td>
<td>Individual</td>
</tr>
<tr>
<td>Association</td>
<td>Association</td>
<td>IndividualOf</td>
</tr>
<tr>
<td>Slot</td>
<td>Class</td>
<td>SlotRelation</td>
</tr>
<tr>
<td>Association</td>
<td>Association</td>
<td>HasSlot</td>
</tr>
<tr>
<td>Facet</td>
<td>Attribute</td>
<td>Facet</td>
</tr>
<tr>
<td>Axiom</td>
<td>Operation, External File</td>
<td>Axiom</td>
</tr>
</tbody>
</table>
Issues Encountered Implementing the UML Profile as a Rose add-in

• Conceptual differences between the KR and UML domains
• Limitations in tool support impacted implementation of ontology modeler add-in, and therefore impacted the profile itself